Computing News

News from the Computing Division
Fermi National Accelerator Laboratory

VOLUME XXIII, NUMBER 2, MAY, 1995

Statement of Direction for operating system and platform support at Fermilab

The Computing Division has proposed and the Directorate has approved the following **Statement of Direction for Operating System and Platform Support at Fermilab**

1. Fermilab will concentrate on supporting computers running the UNIX operating system and personal computers running MacOS, Windows, and Windows NT.

2. To free up resources to improve and expand the support for these systems, we will begin an orderly "phase-out" of support for the VMS operating system. This will begin immediately and will take about three years to complete. This applies to both VAX VMS and ALPHA VMS. The process will be completed well before Collider Run II, but we will try to reduce the impact on the analysis of data from Collider Run I.

3. The lab will support a variety of different computers running the UNIX operating system. Groups will be expected to develop applications which are portable and can run on more than one platform so that we can benefit from competition in our acquisitions. At the same time, it is understood that neither we nor our users can expend the effort to make sure that every application runs on every system. We expect the major activity areas of the lab to choose from the list of supported systems those which are most suited to and most cost effective for their application domain. In most cases, more than one system should be selected to promote competition.

4. Certain control systems, real time systems, and embedded processor applications are not subject to the schedule outlined above. These systems are mission critical real time systems which may be tightly coupled to hardware and are supported by their own in-house groups. Those groups are responsible for planning the evolution of these systems, but should do so in the context of the lab's overall computing strategy as outlined above.

Remarks on the Statement of Direction

The Director of Fermilab and the Associate Director for Information and Technology have asked the Computing Division to develop a plan to expand support for scientific, technical, and administrative computing throughout the lab and to aggressively implement new technologies within the constraints of its present size and budget. In order to achieve this, it is necessary to reduce support in some areas where there is costly overlap or redundancy. This "statement of direction" is a partial response to that challenge.

What's Inside

The "statement of direction" was developed after consultation with many elements of our user community, including spokespeople from the major collider experiments, representatives of major HEP collaborations and heads of all Fermilab Divisions and Sections. It has been stated, in proposal form, at a meeting of all spokespeople of upcoming Fixed Target experimenters, before the Fermilab Physics Council, and before the URA visiting committee. It has recently received the approval of the Fermilab Directorate. While these consultations have been "extensive" they were not "all-inclusive" and many users may be finding out about this for the first time here. It is also fair to say that not everyone who was consulted agrees with this statement of direction. There was, however, general agreement on the necessity to do something along these lines. The main objections concerned the timing.

The rest of this article addresses several issues, including discussion of the time scales involved, some comments about implementation, remarks on what this means for support of various computer vendors, and possible impact on university and other external user groups. It closes with a brief comment on the need to keep an open mind towards even more profound changes that might be necessary in the future.

Motivation

The reasons for moving in this direction are:

1. to achieve better alignment with the mainstream of computing, which now seems to be UNIX, PCs and MACs. Because our software problems are so challenging, we must try to make maximum use of what is available commercially and in the public domain and save our own resources to deal with those problems that are unique to High Energy Physics. We must be positioned to take advantage of new technologies, new software products, and new software methods. Recently, we have been able to use a variety of freeware, shareware, and commercial products to solve some of our problems. The "developmental momentum" is with UNIX and personal computers. We want to tap that source of energy. Other HEP laboratories are also moving in this direction.

2. to eliminate the overhead of supporting two major operating systems for compute-intensive platforms and scientific and technical computing. While there are certainly variations among the UNIX platforms, they tend to be small, a lot of the tools and programs are quite portable, and we are beginning to understand how to deal with them. VMS is quite different and the complications of maintaining both a VAX/VMS and ALPHA/VMS environment produce added work. Finally, to the extent that important activities are not ported from VMS, we are linked to a single vendor whose marketing decisions, pricing policies, and technology directions are completely outside our control and to which there is no alternative.

3. At the same time, the general administrative and much of the technical computing load has migrated (in many cases from VMS machines) to MACs and PCs because of the availability of the applications that people want and need. We have also heard a growing demand from HEP researchers for these applications. The ability to use personal computers as X-terminals offers a real possibility of providing a single platform that allows access to popular commercial applications for word-processing, preparation of presentations, charting, statistical analysis, etc., as well as access to powerful UNIX systems for data analysis and program development.

In order to focus more effort on properly supporting UNIX in a constrained staffing situation and expanding personal computer support, we will phase-out VMS. The people who have been supporting VMS will gradually move on to work on UNIX systems or personal computers.

What this "statement of direction" is not

1. This direction does not represent a negative judgement on the functionality or technical merit of the VMS operating system. That system has served us well over a period of many years. It is, in some sense, the last of the "proprietary operating systems" to survive in the workstation market. The commercial world, which more and more drives computing technology, has moved away from it and we are more or less compelled to follow.

2. This statement does not represent a negative judgement on DEC computing technology. Our original motivation, back in the late 80's and early 90's for moving to UNIX was that UNIX/RISC systems were the most cost effective systems and were certainly more cost effective than the available VMS systems and because we anticipated the move of the market place towards UNIX. Recently, within the last year or two, the DEC ALPHA chip has joined the price/performance leaders. Since that chip can run either UNIX or VMS, we still have an opportunity to take advantage of this powerful device under this new plan. The Computing Division has been evaluating new UNIX platforms and has recently added DEC ALPHAs running Digital UNIX to the list of supported UNIX platforms, where it joins Silicon Graphics, IBM, and SUN. (CD publication DR0010 lists the supported systems. New systems are evaluated when appropriate.) Virtually all programs supported and maintained by the Computing Division on other UNIX platforms have been ported and validated under Digital UNIX.

3. This new direction is not undertaken without considering the impact on university users. There are many aspects of this strategy that will make it difficult to use VMS systems on site. There will be very little support of any kind for VMS. That, after all, is the point of phasing out VMS. However, Fermilab has not tried, and will not try, to tell people what to do at their home institutions. Nevertheless, collaborations may find it less useful to support a whole separate set of software offsite and may find it difficult to do so without any help from the lab. People who have recently invested in ALPHA machines can switch to Digital UNIX if they wish and in that way get aligned with the lab. Some groups may still have to rely on VAXes. While most university groups will be forced to replace all their VAX computers over the next few years anyhow, there may be some groups that sim-

ply cannot find the resources to move away from their VAXes. The number of such groups is small, the resources required to replace the VAXes are not large, and by working together -- the users, the lab staff, and the funding agencies -- we can solve these problems.

Reason for the schedule

There are several practical and programmatic reasons for the timing of this statement and for the 3 year duration of the phase-out of VMS.

The pace of technology change requires us to replace our equipment at least every four years. So, we have included in our long range budget plan for the next three or four years the need to replace many of the key systems that we have relied on over the past few years. This includes almost all of our large fleet of VAX computers. We do not have the budget to move much faster than this. However, if we want to make this transition, we cannot afford to continue to invest in the systems that we are planning to move away from. We cannot move more slowly than this either!

The remaining large users of VMS cycles at Fermilab are CDF and D0. Both groups have plans to reengineer their software and hardware platforms for Run II. This plan allows them to do this in a controlled and phased way and to still have their systems in place well before the beginning of Run II so that bugs can be eliminated and stable performance can be achieved before the run begins. Most of the Fixed Target experiments already have made the transition to UNIX and the platforms they will use are partially in place.

Another reason for a gradual phase-out of VMS rather than a very abrupt one is the need to make sure that ongoing activities, including analysis efforts for data from Collider Run I, are not jeopardized. Many of these analyses are performed by students who have time constraints. Other users may need time to learn the new system and to migrate their work.

On the other hand, it is clear that all efforts directed towards new activities must be consistent with this statement of direction. We have made known for some time that we will not support VMS for the Fixed Target run, either in online or offline systems. VMS will be excluded from Fermilab activities directed towards collider Run II. It is also clear that, in this phase-out, some VMS systems will be eliminated or reduced in size very quickly. We expect the central VMS cluster, FNALV, and many local area VAX Clusters to be reduced in size or disappear in the next year to 18 months. We are currently well positioned to provide UNIX facilities to anyone wishing them through general purpose systems such as FNALU, and CLUBS, or dedicated UNIX systems such as exist for both CDF and D0. All our reconstruction and Monte Carlo production facilities have been UNIX systems now for many years. We are also planning to improve our personal computer support for users that move in that direction. We will discuss those plans in the near future.

Implementation

We have done much work to understand the scope of the migration problem. But to really understand these problems we need the full participation of our users. We cannot really expect people who are under intense pressure to run the experimental program and get the data analyzed to focus on these issues if there is no clear direction or timetable. We believe that we understand how to go about these kinds of migrations because we have accomplished several of them. In many cases, there are successful efforts in place already on UNIX machines because some of the users made the transition to UNIX during the period where it had the big price/performance advantage. The users that remain on VAXes will need to be retrained but the facilities are already there on UNIX and well-tested. We will work to establish schedules for the downsizing and eventual elimination of each system and we will be as aggressive as our resources permit. We have been working on all these issues for a long time and are in a good position to address them.

We will use the following general approach to the migration problem

1. As we have done in the past, we will set up a special working group with Computing Division members, key users, and representatives of major lab organizations to handle the migration issues relating to general lab scientific, administrative, and technical computing and to the Fixed Target program. We expect VMS activity from this segment of our community to be at a tiny level about 18 months into the migration effort. This group will also work to assist CDF and D0 physicists working on Run I data and will try to maintain coherence between its efforts and those aiming for collider RUN II.

2. CDF and D0 will be involved not only in migration but in reengineering their computing for RUN II. We plan to assist them in this effort.

3. We will work with external users to try to help them, to the extent that we can, cope with any issues that impact them.

Possible future changes

One important lesson that we all should be able to agree on from our recent experiences in computing is that the pace of change is now so fast that we have to plan for it. None of us can promise the user community that we will not have to undertake a new round of dramatic changes in the next ten or even in the next five years. We have to build into our computing strategies agility, resiliency, and flexibility. We must pick widely accepted standards and then hope that they continue to satisfy our needs and to be commercially viable as well for some reasonable time. But, since there are no guarantees, we have to be easily able to switch to something new. We have to build our software and hardware strategies and architectures to accomplish this.

Joel Butler, butler@fnal.gov

First VMS Migration Task Force formed

The first of the several task forces mandated to implement the statement of direction described above is now being formed. It will work with general laboratory technical, scientific, and administrative users as well as experimenters from the Fixed Target program to accomplish the transition to the supported UNIX platforms or to personal computers. I have asked Judith Nicholls to run this task force. The membership is still being determined but it will include key members of several departments within CD and it will have access to many of the resources of those departments to solve specific problems that arise. Of course, as with all past migrations, success will require strong participation and effort on the part of the user community.

The abbreviated charge to the task force is:

"To identify problems associated with migrating users from VMS-based systems to the most appropriate supported platform—either UNIX systems or personal computers—and to assist users in solving them."

"To be responsible for user training and for documentation in support of this direction."

"To collect user input and transmit it to other organizations which are providing support for UNIX systems and personal computers so that ongoing support can adjust to the needs of these new users."

Since user training and education falls within the mission of this task force, it will also be responsible for assisting physicists from CDF and D0 who are working on RUN I data and who want or need to change to UNIX.

The task force will begin its activities immediately and will be contacting users in about a month.

Joel Butler, butler@fnal.gov

Serial media

Below is a summary of the report of the Serial Media Working Group (see the Computing News of June 1994). These recommendations have been submitted to the Computing Division. This is the first step in the process of defining our approach to serial media for the next several years. The next step will be to have an open strategy meeting which will consist of two parts. In the first part, members of the working group will present a more detailed explanation of their conclusions and the reasoning behind them. They will be available to answer questions from users. In the second part, users will be able to comment on the recommendations and the impact they will have, if implemented, on their activities. Discussions of alternatives are welcome. While all users who are likely to be affected by these decisions are welcome to attend, we strongly encourage those from groups who will be taking data in the next fixed target run to participate since they will be strongly impacted by whatever strategy is finally chosen.

Tape Technology

Over the past few months the Serial Media Working Group has been examining the current state of tape storage technology with the aim of identifying a suitable replacement for 8mm tapes. A primary concern in this quest has been the reliability of any such replacement. We have not found a technology that we feel currently warrants a large scale change over from 8mm. However, in the course of the investigation we do feel we have gained some insight into reliability issues with 8mm drives.

We believe, based on information from Exabyte, Sony, studies at the National Media Labs, and our own experience at FNAL, that we are significantly increasing the failure rate of 8mm drives through the use of video grade tapes. The video grade tapes, including the currently stocked Fuji tapes, are believed to cause excessive wear to the drives both through abrasiveness of the tape and decay of the tape substrate with age. Lower quality shell construction of the video grade tapes is also believed to contribute to commonly seen mechanical failure modes. While we have not yet seen evidence of large scale tape lifetime problems at the lab, it is clear from our investigations that we should not expect the archival lifetime of video grade tapes to exceed 5 years.

Based on the information we have found we are recommending that the laboratory begin stocking data grade tapes as soon as possible. Users should plan on shifting to the data grade media as soon as it is available.

We do not recommend that large scale copying of existing data sets to data grade tapes be done. There are, however, some limited sets of frequently read data that are likely to benefit from residing on data grade tapes. We have, therefore, recommended that the division increase the capacity of the copy facility to accommodate transfer of reasonable size data sets to data grade tapes.

Systems that can be isolated from existing video grade tapes are expected to show the greatest benefit from this change. Any systems on which the type of media can be practically restricted should do so. Drives which must be exposed to a mix of video and data grade tapes can expect to see the benefits of data grade tapes reduced accordingly.

Our information indicates that data grade tapes can be purchased by the laboratory for $4.50-6.50 each. This compares to $2.70 each for the current stock of video grade tapes.

We expect the laboratory to evaluate the results of the media change over six months after the introduction of data grade tapes. At that point a decision will be made whether to continue allowing video grade tapes on common access systems and whether such tapes should continue to be accepted for vaulting at FCC.

A complete text of the recommendation can be found at `http://d0sgi1.fnal.gov/data_tapes.html`

Recommendation on New Serial Technology

Based upon studies over the last several months, the Serial Media Working Group has made the following recommendations to the Computing Division management regarding tape technology.

At this time we feel it would be premature for the laboratory to adopt another tape drive standard over 8mm technology. Within the constraint that any new technology be generally available in time for the upcoming fixed target run, we find that the DLT technology from Quantum is the only viable alternative to 8mm. However, because of the expense of the equipment and media, uncertainties about the drive reliability, and prospects of other attractive technologies appearing within the next year, we do not recommend that the lab replace 8mm technology with DLTs for general purpose data distribution at this time. We recommend that the main line of support for serial media at the lab for the next 12-18 months be 8mm technology using data grade media.

In spite of the higher cost and reliability uncertainties of the DLTs, we recognize that some groups will find this technology very attractive due to its higher capacity and bandwidth. We recommend that limited support be provided to those groups which can demonstrate a clear need for the higher capabilities of the DLT drives. We recommend that groups which feel they can demonstrate such need discuss the possibility of isolated DLT support with the Computing Division soon. Any group expecting to ask for such support should be prepared to make compelling arguments regarding their need for the increased capacity and bandwidth. Any group which is allowed to adopt DLT technology should not expect the lab to provide data transcription from high density to low density media.

The above recommendations should not be taken to indicate that we feel the laboratory should remain standardized on 8mm tape for the next 3-5 years. We believe there will be several technologies available within the next year which should be seriously considered. However, except for DLTs, it is not likely any of them could be implemented in time to offer a solution for fixed target experiments. Nor should the recommendation be construed as excluding a large scale commitment to DLTs in the future.

One of our primary concerns in evaluating new technology has been the issue of reliability. We have found not only that vendor-supplied information is inadequate to judge reliability of new technology, but that our own record keeping at the lab does not allow us to do a sound analysis of the technology currently in use. Therefore we are recommending that the lab institute a quality assurance program that will allow the Computing Division to accurately evaluate the performance of any serial technology that is deployed. This program should include careful tracking of both usage and repair history of drives, as well as detailed examination of selected problem reports to ensure our understanding of recurring failure modes. Users should be prepared to cooperate with requests for information to aid in this tracking effort.

The full text of this recommendation can be found at `http://d0sgi1.fnal.gov/new_technology.html`

Mike Diesburg, diesburg@fnal.gov

Accounting and CPU ratings

The rapid development of central processing units (CPUs) on computing systems has created a need to understand the relative speed of many computers in order to better understand how well the computers are being used, to help plan for added capacity, and to plan for the removal of old computers. The pace of change in the computing industry makes this a difficult job. The mix of computing jobs on systems also tends to make such a measurement difficult.

We have selected the code **tiny**, a Monte Carlo generator and pattern recognition and track fitting program, to serve us in this capacity. The code has the following attributes:

1. It runs on all platforms
2. It is easy to run and port to new machines/platforms
3. It gives consistent results
4. It uses HEP fortran code.
5. It is available to all to use

It has to be noted that **tiny** is not representative of any individual HEP reconstruction code nor is it able to measure the performance of a specific computer under normal loads. Competitive acquisitions of computing systems by Fermilab use broader benchmarks which are more representative of the computing system that is required.

The Computing Division has chosen to use this program to scale each CPU accounting record in order to provide a uniform measure of CPU power. The ratings that are being used to provide the computer usage information shown in the tables published in document TN0082. The UNIX farms are essentially the only machines which continue to use their old ratings. This is to maintain some continuity with many years of accounting records on those machines.

A more detailed description of the program, as well as how it compares to other benchmarks such as SPECint92 and SPECfp92 and to user codes such as E665 and CDF, is found in the Computing Division library as TN0082.

Steve Wolbers, x3950, wolbers@fnal.gov

Applications

TRANSPORT to be available as part of Fermitools

Transport is a computer program used to design charged particle beam transport systems. Beam line components which can be represented are drifts, bending magnets, quadrupoles, sextupoles, octupoles, solenoids, travelling wave accelerating cavities, steering magnets, and plasma lenses. The effect of the beam line on a charged particle trajectory is represented by first-, second-, and third-order matrices. Beam phase space dimensions and floor coordinates may also be calculated. The beam phase space can be specified as a matrix of variances and covariances or in accelerator parameters. Its center can deviate from the reference trajectory along which the magnetic components are placed.

Transport will do fitting, allowing the simultaneous variation of up to 20 selected parameters to satisfy imposed constraints. Misalignments and errors may be simulated and their effect evaluated. In conjunction with a plotting program such as **Topdrawer**, plots can be made of the beam ellipse, any matrix element against accumulated length, and the floor layout of the beam line, including three-dimensional representations of the magnets.

Input can be in either the **Mad** compatible notation of Christoph Iselin or in a strictly numerical format. The parameters can be in any units desired. Physical parameters and constrained quantities can be related by algebraic expressions.

Transport is now available as part of the **Fermitools** program - from anonymous ftp at `ftp.fnal.gov` under the `/pub` directory. Versions are available for VMS and IRIX. Questions can be addressed to`transport_support@fnal.gov`.

Dave Carey, x 3639, davec@fnal.gov

C D

UPS v3_3 features

After an extensive design, development, and testing process, the UNIX Application Support Group has released UPS v3_3 as current in the KITS area. UPS v3_3 has several new features. Most notable is the support for “extended flavors”. Extended flavors refers to the additional operating system and hardware related features for which a product version has been specifically designed. For instance, there are currently two incompatible flavors of IRIX at Fermilab. These are often referred to as IRIX 4 and IRIX 5. Software compiled on IRIX 5 will not execute correctly on machines running IRIX 4.

One of the simplest forms of extended flavor support is the automatic recognition by **setup** of the release of the operating system at execution time. **setup** will then set up the best match flavor as the product requested was declared. Most often these flavors would be declared as IRIX and IRIX+5. However, extended flavor support does not stop there.

Products could be declared as `IRIX+5+mips2+debug`, for instance. Or it may be desirable to set up the `mips2+debug` flavor of the product without failing if the `mips2+debug` flavor of the requested product version has not been declared. If the environment variable `UPS_EXTENDED_FLAVOR` is set to `mips2+debug` and `setup abc` is issued, **setup** will set up the “current” version of product`abc` having a flavor matching the operating system and best matching the operating system release and matching as many of the other (`mips2, debug`) as possible.

Notice that no meaning is associated with extended flavor fields beyond the operating system release. The order in string `UPS_EXTENDED_FLAVOR` will determine the relative importance of the individual field. The order of the fields is not related to the order in the declared flavor of the product version.

However, because there is no meaning associated with the additional fields, **setup** cannot set up products with declared flavors in addition to those requested. For instance, a product declared with flavor `IRIX+5+debug+mips2+magic` would not be set up, because **setup** cannot determine if `magic` is exclusive. Indeed, in the proceeding example, only the following flavors of `abc` current could be set up on an IRIX 5.0.2 system:

IRIX+5.0.2+mips2+debug
IRIX+5.0.2+debug+mips2
IRIX+5.0.2+mips2
IRIX+5.0.2+debug
IRIX+5.0.2
IRIX+5.0+mips2+debug
IRIX+5.0+debug+mips2
IRIX+5.0+mips2
IRIX+5.0+debug
IRIX+5.0
IRIX+5+mips2+debug
IRIX+5+debug+mips2
IRIX+5+mips2
IRIX+5+debug
IRIX+5
IRIX+mips2+debug
IRIX+debug+mips2
IRIX+mips2
IRIX

The instance which will be set up will be the first one encountered while searching the **ups** database(s) in `$PRODUCTS` in the above list. Few products are expected to require greater specificity than the first digit of the operating system release.

Also, methods exist to override or extend `UPS_EXTENDED_FLAVOR` on the **setup** command. And, of course, one could specify the exact flavor required in the `-f` *`flavor`* of **setup**.

Additional Features (primarily for product developers, maintainers, installers)

ups declare and **modify** now support the **-Z** option allowing them to “guess” the database into which to declare the product. Both will prompt for permission before proceeding.

The new **ups depend** command will list product dependencies. It is coded in **perl** and requires the **shells** product to execute correctly.

Several enhancements to catch and prevent **ups** database corruption have been added.

By the time you read this, UPS v3_3 will likely have been replaced with UPS v3_3a, v3_3b, ... as bugs are eliminated. There is a commitment to provide bug fix releases for UPS v3_3 while the implementation of UPS v3_4 is going on. The goal is to make a beta-release of UPS v3_4 available in mid-June.

Chuck DeBaun, x3522, debaun@fnal.gov

UCM - a layer above RCS for management of experiment code

A new product called **UCM** (UNIX Code Management) is now available through KITS and has been installed on FNALU. It contains tools which experiments can use to put together a UNIX-based code management system, similar to existing systems which use the VMS utility, **CMS**, to manage source code. **UCM** was developed as a collaboration between the UNIX Code Management Working Group and Experiment E831 to meet the needs of managing experiment code from a UNIX repository.

The **UCM** package contains two utilities, **uvma** and **uvmi**, to create and manage directories of **rcs** files called UVM (UNIX Version Management) libraries. These utilities provide the essential features of **CMS**, including access control (reserve, replace, and fetch), information gathering, and commands to tag specific versions of a file. This latter feature can be used to create the equivalent of a **CMS** "class", which is used to record particular release versions of a software package.

The commands provided by the **uvma** (UVM Access) utility deal with all aspects of library creation and maintenance. These include: library creation; element creation, reservation, replacement, and fetch; adding and removing version tags; and commands to change library and element descriptions.

The **uvmi** (UVM Inquire) commands extract various types of information about a uvm library and its elements. The types of information that can be extracted include: a library description; lists of reserved elements; a summary of library activity (i.e., when were library elements created or changed and by whom); a more detailed list of changes to specific elements, including comments and an element description; information about tagged revisions; and a summary of differences between "release" versions. To help maintain a consistent set of source files that reflect the list of include files they depend on, the **UCM** package also contains a command called **uvmbld** which can be used to construct suitable **makefiles** from UVM libraries.

Finally **UCM** contains a program called **KPP** which is intended to replace part of the functionality of the program **expand**. **expand** was written by the CDF collaboration to provide a preprocessor for Fortran source code which is used to handle machine dependent code and provide a platform independent means of handling include files. **KPP** provides a language insensitive preprocessor which can be used to standardize include-file references and create conditional text. The command formats are the same as **expand**.

Peter Kasper, x3732, kasper@fnal.gov

Jonathan Streets, x3629, streets@fnal.gov

Release of ximagetools v2_1a

Overview

The UNIX Application Support Group has released a new version of the **ximagetools** product which replaces the **xli** image viewer with the much nicer **ImageMagick** tools, which include:

display — an image viewer similar to **xv**
convert — an image conversion utility
animate — a multi-image animation player.
import — grab an X window screen image

There is a symbolic link from **xli** to **display**, so that existing configurations which used ximagetools v1_0 or v1_1 will still work unmodified.

Image Formats

The ImageMagick tools support (read and write) the following image formats (except those noted read-only):

AVS	AVS X image file.
BMP	Microsoft Windows bitmap image file.
CMYK	Raw cyan, magenta, yellow, and black bytes.
EPS	Adobe Encapsulated PostScript file.
EPSF	Adobe Encapsulated PostScript file.
EPSI	Adobe Encapsulated PostScript Interchange format.
FAX	Group 3.
FITS	Flexible Image Transport System.
GIF	Compuserve Graphics image file.
GIF87	Compuserve Graphics image file (version 87a).
GRAY	Raw gray bytes.
HISTOGRAM	
IRIS	SGI RGB image file.
JPEG	
MAP	Red, green, and blue colormap bytes followed by the image colormap indexes.
MATTE	Raw matte bytes.
MIFF	Magick image file format.
MTV	
NULL	NULL image.
PCD	Photo CD.
PCX	ZSoft IBM PC Paintbrush file.
PICT	Apple Macintosh QuickDraw/PICT file.
PNM	Portable bitmap.
PS	Adobe PostScript file.
PS2	Adobe Level II PostScript file.
RAD	Radiance image format.
RGB	Raw red, green, and blue bytes.
RLE	Utah Run length encoded image file; read only.
SUN	SUN Rasterfile.
TEXT	raw text file; read only.
TGA	Truevision Targa image file.
TIFF	Tagged Image File Format.
VICAR	read only.

VID	Visual Image Directory.
VIFF	Khoros Visualization image file.
X	select image from X server screen.
XC	constant image of X server color. Specify the desired color as the filename.
XBM	X11 bitmap file.
XPM	X11 pixmap file.
XWD	X Window System window dump image file.
YUV	CCIR 601 4:1:1 file.
YUV3	CCIR-601 4:1:1 files.

Memory usage

The ImageMagick utilities like to use lots of virtual memory, especially on large image formats like multi-page tiff and mpeg images. We recommend against using ImageMagick's **convert** on multi-page tiff files. Instead, use **import** to screen-grab xtiff screen images.

Other programs

The **xfig** figure editor is still included with **ximagetools**, which provides a nice tool for drawing various sorts of diagrams.

The **xtiff** and **mpeg_play** viewers are included, which have much faster startup times than ImageMagick's tools for tiff and mpeg files, since ImageMagick converts to its own internal format before displaying.

Marc Mengel, x8256, mengel@fnal.gov

CERN Library news

DI3000 support and CERN libraries

As of the v94b release of the CERN library, we have dropped support of the DI3000 graphics package for all platforms. That means there is no DI3000 version of the **HIGZ** library and no DI3000 version of **PAW**. For all prior releases on VMS systems, the CERN setup script defined a symbol **paw** that pointed to the DI3000 version of the **PAW** executable. With the v94b release, that symbol now points to the **PAWX11** executable. There are several consequences to that change:

- Workstations that do not support or at least emulate the X11 client/server protocol can no longer be used for **PAW** sessions.
- The opening dialog for selecting a workstation type is slightly different. The details are discussed in both the **HPLOT**/**HIGZ** manual and in the **PAW** manual.
- The metafile command works as described in the **PAW** manual. This was not the case for the DI3000 version of **PAW**.
- The only supported form for hard copy is postScript. Normal postscript (PS) in both portrait and landscape orientation and encapsulated postscript (EPS) are available. Again, this is discussed in the **PAW** manual.

We have been considering dropping support for the DI3000 versions of **HIGZ** and **PAW** ever since the original vendor was acquired by another company. The support level for DI3000 by the new company appears somewhat unenthusiastic, as several previously supported device drivers have been dropped. The wide range of supported devices was one of the major attractions of DI3000 but, as that range narrowed, much of the attraction has vanished.

Calling Fortran routines from C programs

Historically, the bulk of the CERN library was written in Fortran although a substantial portion of the newer, non-mathematical code is being written in C. From time to time, the question of calling those Fortran routines from programs written in C arises. There are a number of troublesome details that need attention when attempting that. One of the CERN developers has written a tutorial article on that subject and we have extracted it and made it available through the PAT WWW web server along with several other articles, bug reports and announcements. The URL for the PAT web server is

`http://cdibm.fnal.gov/cern.html`

That page also includes a link to another WWW server at GSI in Darmstadt where there is an article on a C macro generator that does most of the job for you. Users writing in C who wish to use any of the CERN library routines or, for that matter, any external routines written in Fortran, should at least look at these articles. The information they contain could prevent a lot of unnecessary debugging and the accompanying time loss and frustration.

John Marraffino, x4483, marrafino@fnal.gov

Flint v3_01

The UNIX Application Support Group, has recently made Flint v3_01, the latest version from the vendor available in both AFS space and in the UNIX KITS area. The supported flavors are IRIX 5 and SunOS 5 (Solaris 2).

Starting with this release, users must either use the version of **flint** in AFS space or obtain a local copy from the KITS product database. All versions will point back to a single license manager. Previously, there was a requirement to NFS mount the **flint** product, and only IRIX 4 was supported. Support for the NFS mounting of **flint** will end June 1, 1995 as our license for the IRIX 4 version will expire at that time

Etta Burns, x8300, ettab@fnal.gov

MXYZPTLK: a C++ class library for AD/DA

If you need to calculate derivatives of complicated functions and find yourself either taking finite differences or writing the derivatives algebraically and then translating the expressions into source code, you may want to consider using automatic differentiation (AD). AD exploits the classic theorems of differential calculus to propagate information about derivatives through arithmetic operations. In this way, derivatives of a function can be calculated using the same program that calculates the function itself. Because no approximations are made, derivatives are calculated with machine accuracy, avoiding the errors inherent in finite differences, an especially important consideration when higher order derivatives are required.

MXYZPTLK is a library of C++ classes — or "objects" — for performing automatic differentiation. Originally written at Fermilab in 1989, with a "User's Guide" provided in 1990, it has undergone repeated correction and improvements over the last six years. It was formally announced outside Fermilab in *Automatic Differentiation of Algorithms: Theory, Implementation, and Application* (SIAM Press, 1991) and has been used in a variety of contexts. **MXYZPTLK** remains the only implementation of AD which exploited object oriented techniques (in C++) from the beginning.

A simple example illustrates the way the library could be used to calculate an "error sensitivity matrix" for estimating Cartesian (x, y, z) position errors from a measurement of polar (r, θ, Υ) coordinates.

```
#include "mxyzptlk.rsc"

main( int, char** argv ) {
const double d2r = M_PI / 180.0;
MatrixD M( 3, 3 );
Jet::Setup( 3, 1 );

coord r     (      atof( argv[1] ) ),
theta ( d2r*atof( argv[2] ) ),
phi   ( d2r*atof( argv[3] ) );
Map   position;
position.SetComponent( 0, r * sin( theta ) * cos( phi ) );
position.SetComponent( 1, r * sin( theta ) * sin( phi ) );
position.SetComponent( 2, r * cos( theta )              );

M = position.Jacobian();
cout << M << ""n"n" << M.inverse() << endl;
}
```

With these few lines, the program accepts the three arguments r, θ, and Υ (with angles in degrees) from the command line and writes both the matrix, $\underline{\underline{M}} = \partial(x, y, z)/\partial(r, \theta, \Upsilon)$ and its inverse on the standard output. The variable types `coord`, `Map`, and `MatrixD` are defined in the `mxyzptlk.rsc` header file. They are made available to the program upon linking with the **MXYZPTLK** library.

A second example shows a more sophisticated application: evaluating the inverse of a `function` at a point specified on the command line.

```
#include "mxyzptlk.rsc"

main( int, char** argv ) {
Jet::Setup( 1, atoi( argv[1] ), 1 )
double a = atof( argv[2] );
coord x(0.0);
Jet   f;
f = sin( x + x*x + x*x*x );
cout << f.Inverse()( &a ) << endl;
}
```

The `Jet` variable, `f`, implements the function, $f(x) = \sin(x + x^2 + x^3)$, whose inverse is constructed up to an order, in the coordinate, x, that is stipulated on the command line (and fed into the preliminary ::`Setup` routine). In this example, f^{-1} is expanded about $x = 0$ but the reference point could have been selected arbitrarily. Other variations are easily constructed; for example, (a) the function to be inverted could be a multi-variable map, $f : R^n \to R^n$ (b) it could be a complex valued function, or (c) the inverses of several functions could be constructed simultaneously, stored in `Jet` variables and manipulated as functions.

MXYZPTLK has been used at Fermilab, in conjunction with a second library, **BEAMLINE**, to construct and analyze Poincare maps associated with accelerator physics. A "tracking program" which uses coord objects, rather than simple double precision variables, will not only do tracking but simultaneously compute all the derivatives of the one-turn map up to whatever order de- sired. That information can then be processed, using LieOperator objects from the library, and other tools, to obtain useful physical quantities, such as tune spreads, resonance strengths, or periodic orbits and their separatrices. (This is the "differential algebra (DA)" part of the library.) By including strengths of magnet elements within the family of `coord` objects, sensitivities to their variations can be calculated as well. (For example, by using `coord` variables in tune correction circuits and computing the tune with no correction, a function is obtained relating tunes to quad settings, which can then be inverted, as was done in the example above, to write a simple tune-fitting procedure.)

MXYZPTLK is under continual development (as is **BEAMLINE**), but anyone who wishes to examine it in its current state, can obtain it using anonymous ftp as follows:

```
ftp calvin.fnal.gov
cd pub/outgoing/michelotti/MXYZPTLK
mget *
```

You will receive a compressed tar file, `mxyzptlk.tar.Z`. Using a UNIX-based computer, do

```
uncompress mxyzptlk.tar
tar xf mxyzptlk.tar
```

Subdirectories `src`, `include`, `demos`, `lib`, and `docs` will be created and filled. Regrettably, the User's Guide, unchanged since 1990, while not useless, is out of date. Nonetheless, the `demos` area contains a few programs that may help to get a feel for how the software works. The author is working with the Fermitools Working Group to make **MXYZPTLK** available as part of the Fermitools Program within the near future.

Leo Michelotti, Accelerator Division, x4956

Distributed Computing

Distributed Computing announcement

The Distributed Computing Department is pleased to announce the availability of GG0016, *TCP/IP based Macintosh Remote Computing over Standard Dialup Connections*, in the Computing Division Library on WH8NE.

This document is a step by step guide to the installation of the MacTCP and InterSLIP extensions and selected TCP/IP applications on the Macintosh in order to implement TCP/IP-based Macintosh remote computing. A disk with the required software is also available from the Computing Division Library. In order to receive a copy of the disk, you must first demonstrate that you are licensed for MacTCP.

In addition to the above document and disk, the Fermilab stockroom is also stocking "Apple Macintosh to Modem cables with hardware handshaking" (1410-4000) and the book, *Internet Starter Kit for the Macintosh*, by Adam C. Engst (1307-0600), which includes a MacTCP license. Purchase of this book from the stockroom or one of the local bookstores is a quick and easy way to ensure that you are licensed for MacTCP.

Once the software is installed on the remote (i.e., home) Macintosh, you can enjoy many advantages over the traditional ANSI terminal and modem combination:

- Direct access to the TCP/IP systems on the Fermilab Network.
- Direct access to off-site TCP/IP systems (with the appropriate access and/or accounts).
- A workstation-like computing environment including multiple simultaneously active windows and applications.
- Easy and rapid file transfer between local and remote systems.

A cautionary word — Macintosh based remote computing using Fermilab and ESnet resources is subject to the same mission related restrictions as on-site computing using Fermilab computing resources. Macintosh based remote computing is **not** a license to roam inappropriate areas on the Internet at DOE's expense. If you desire or wish to explore such areas of the Internet, you would be advised to contact one of the several local commercial Internet Service Providers (ISP's) in the Chicago area which offer SLIP and/or PPP access and establish an account with them.

Keith Chadwick, x2498, chadwick@fnal.gov

Training

Tutorial series on OOP in C++

In the first week of March we started a tutorial series on object-oriented programming (OOP) in C++. These are aimed at a "FORTRAN programmer" level, which assumes that the attendees want to learn C++ but know nothing about it or, more generally, about OOP. Each session is informal and extemporaneous; I simply do not have time to prepare a polished "course." Although there is an agenda, it is flexible. In particular, the direction that a session takes can be influenced by the questions that people ask.

We will not cover all bases at once. I plan to aim these discussions toward scientific applications (esp., accelerator/detector/high energy physics) as quickly as possible. As time goes on, this will eventually require talking about physics as well as programming, because the way we program -- and the objects we need to design -- will depend on the physics problems we are trying to solve. Ideally, past a certain point, we should be able to jump back and forth between the two almost seamlessly.

The OOP/C++ sessions are held in Curia II on Wednesdays at 11:30. I try to hold them to 30-40 minutes apiece. Anyone who is interested is more than welcome to attend. The one requirement is that you are serious about learning OOP/C++. Put another way, the only people who would not be welcome are those who prefer engaging in lengthy, abstract, and generally sterile debates about OOP paradigms, languages, etc. This is a series in specifics, not abstractions. It is for those who have made their key decisions, at least tentatively, and now want to get on with the job of programming.

We have just reached the point of examining the structure of an "object," so a newcomer should still be able to pick up the thread quickly. If you are interested and would like to be put on our mailing list, please send me email at michelot@calvin.fnal.gov.

Leo Michelotti, Accelerator Division, x4956

A Guide to Computing Division Services

TELEPHONE: (708) 840-xxxx

Request forms: Newsgroup fnal.forms, INFO, or CD Library, WH8E

New accounts, password changes, WH8X........x8118

Computing Division (Voice Mail)........x3206

FAX 8th Floor........x2783

Feynman Computing Center FAX........x3785

HELP DESK........x2345

DATA CENTER SERVICES........x2746

Feynman Computing Center........x2746

DATA COMMUNICATIONS

Trouble Reporting........x3239

ONLINE SYSTEMS

Trouble Reporting/Tech Liaison, WH8E........ols@fnal
(On call pager through operations)

COMPUTER HARDWARE

Trouble Reporting........x4373

EQUIPMENT SUPPORT GROUP

Repair Inquiry........x2688

EQUIPMENT LOGISTICS SERVICES

Inquiry Service Counter........x3447

DIVISION MANAGEMENT

Division Head, Joel Butler........x3148

Deputy Division Head, Vicky White........x3936

Associate Division Head, Irwin Gaines........x4022

Assistant Division Heads

Gerry Bellendir........x3930

Judy Nicholls........x3989

Al Thomas........x3064

Adam Walters, PREP Equipment Manager........x2696

COMPUTER PROTECTION PROGRAM MANAGER

Irwin Gaines........x4022

COMPUTER DIAL-UP NUMBERS AND PORT SELECTOR CLASS CODES

Dial-In (up to 28800bps V.FC/MNP10)840-8134

Callback (up to 28800bps V.FC/MNP10)........840-8555

FNALV VAX Cluster........FNALV

FNALD VAX Cluster........FNALD

Internet Protocol........IP

Terminal Server........LAN

DECserver, 580........DS550

Change of Address

If you have changed your address, or are not on *Computing News* mailing list and would like to placed on it, or would like to be removed from the list, please complete this form and mail it to the Editing Staff.

Action to be Taken (check one):

Deletion ❑ Addition ❑ Correction ❑

Return completed form to:

Pat Haring
FERMILAB - MS 120
P.O. Box 500
Batavia, IL 60510

Address Information (Include Zip Code, please)

____

____

____

____